

FIG. 1

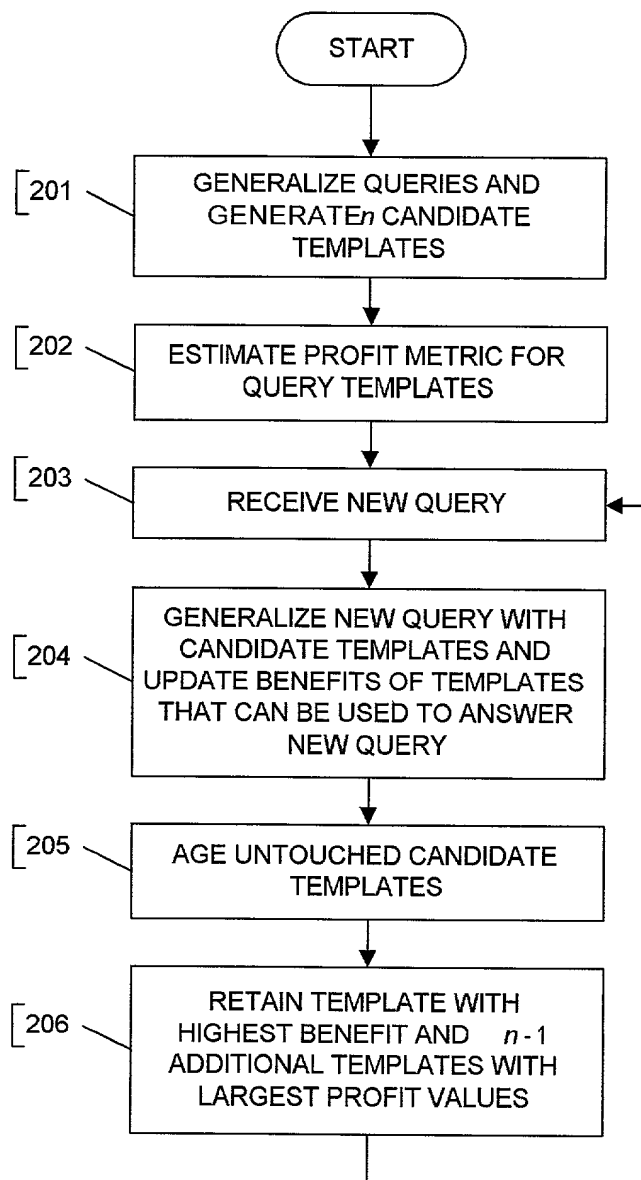


FIG. 2

```

ChooseCandidates( $q, CT$ ) {
  /*  $CT = \{ct_1, \dots, ct_n\}$  */
   $NT = \emptyset$ 
  for each  $ct_i$  in  $CT$ 
     $nt_i = \text{CompPairTemplates}(ct_i, q)$ 
    if ( $nt_i = ct_i$ )
      /*  $q$ : specialization of  $ct_i$  */
       $b(ct_i) = b(ct_i) + c(q)$ 
    else if ( $nt_i \in NT$ )
      /*  $nt_i$ : template exists */
       $b(nt_i) = \max(b(nt_i),$ 
                      $b(ct_i) + c(q))$ 
    else if ( $s(nt_i) < S$ )
       $b(nt_i) = b(ct_i) + c(q)$ 
      add  $nt_i$  to  $NT$ 
  age each untouched  $ct_i$ 
   $NT = NT \cup CT$ 
  if ( $q \notin NT$  and  $s(q) < S$ )
     $b(q) = c(q)$ 
     $NT = NT \cup q$ 
   $CT = \{\text{template with highest}$ 
         $\text{benefit in } NT\}$ 
  choose  $(n-1)$  additional
    templates with largest
    values of profit  $p(t)$  in  $NT$ 
  return  $CT$ 
}

```

FIG. 3

```

Revolution( $AT, CT$ ) {
  /* compute  $CT' \subseteq AT \cup CT$  */
  /* for admission */
  sort the  $t_i$ 's using
     $p(t_i) = \frac{b(t_i) - c(t_i)}{s(t_i)}$ 
   $CT' = \emptyset$ 
  repeat
    add the highest ranked
      remaining  $t_i$  that can
      fit in the available
      cache space to  $CT'$ 
    adjust free space to
      reflect  $s(t_i)$ 
    adjust benefits, costs,
      sizes of unselected
      templates in  $CT \cup AT$ 
    resort
  until (no more templates
        can be added)
   $CT'' = \text{template } t \text{ in } CT \cup AT$ 
    with highest value of
     $b(t) - c(t)$ 
  if ( $b(CT'') \geq b(CT')$ )
    return  $CT''$ 
  else return  $CT'$ 
}

```

FIG. 5

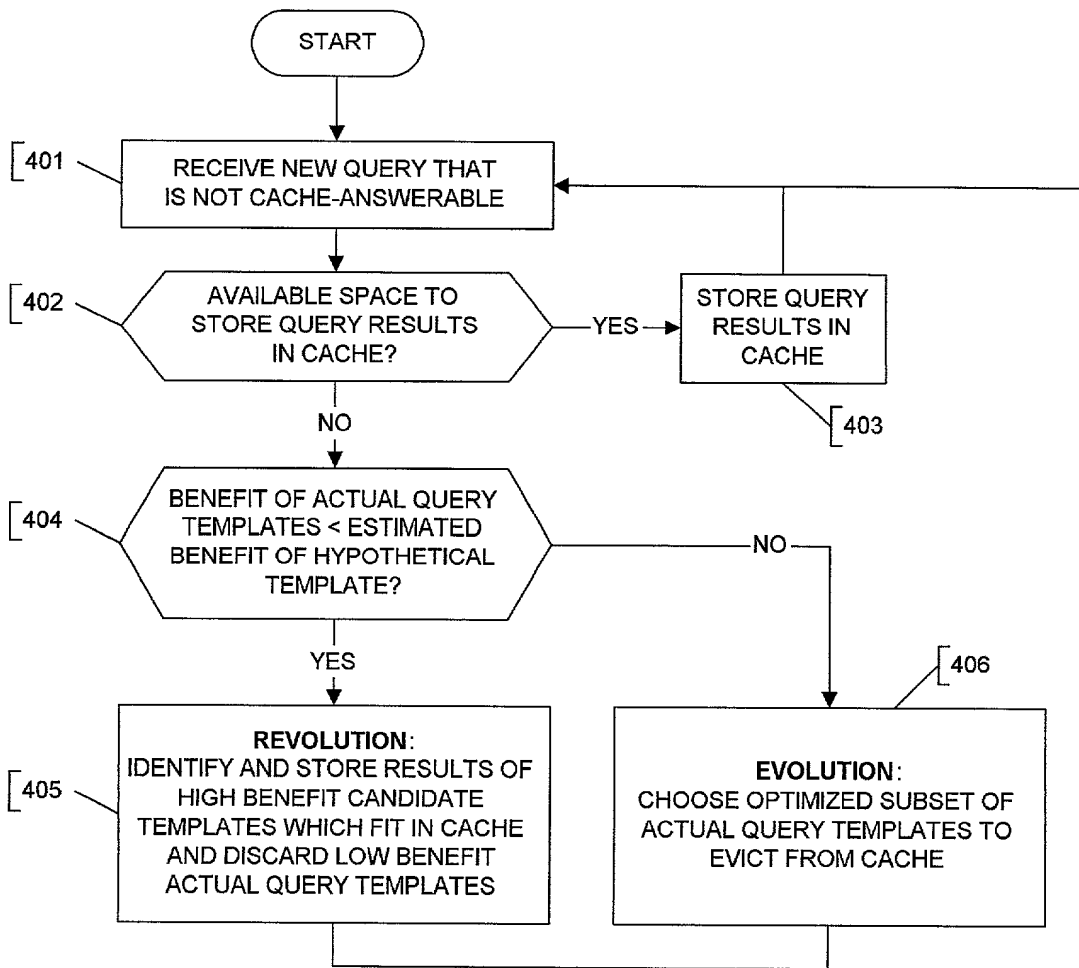


FIG. 4